Patent claims

1. Method for automatically generating software in which the properties of an application made possible by the software are modelled in abstract form and then mechanically converted into software for this application, while the execution of the application influences a technical system optionally made up of a plurality of systems, **characterised in that** at least one of the following additional elements is generated in a totally integrated multi-objective form from the modelled description of the application together with the application source code or together with the source information suitable for producing this source code, namely:

- software for visualising and/or logging and/or remotely monitoring/operating the application and/or the technical system;
- software for simulating the application and/or the technical system;
- software and/or information for communicating within the application and/or with other systems and/or between split systems and
- documentation for the user and/or the programmer.

2. Method according to claim 1, characterised in that in addition to the software for simulating the application, software for counter-simulating the technical system which is influenced by the application is also generated.

3. Method according to claim 1, characterised in that the application is modelled by one or more modules, additional information and possible instancing tables,

wherein

- one module advantageously contains a partial problem of the application,
- the additional information contains information such as text, images, visualisers and type definitions to which reference is made within the modules, and
- the instancing tables contain information which cannot be deposited directly in the module itself in the event of multiple instancing of the modules and which cannot be generated mechanically either, such as hardware addresses, for example.

4. Method according to claim 3, characterised in that a module is totally defined by the following sets of definitions:

- node definitions for distributing the application to physically separate hardware systems coupled by data technology (split systems),
- sub-module definitions for instancing (tying-in) sub-modules,
- element definitions for combining all the data as well as hardware and communication inputs/outputs of a module,
- man-machine interface definitions for defining all the components required within a module for producing an interface for the user and
- function definitions consisting of a number of functions of a module.